

《程式設計概要》

試題評析	本試題四題全部是C++程式題，看完程式後寫出答案，所以著重語法概念。第一題使用堆疊的概念處理字串，使用pop的規則處理字串的前後字元。第二題主要測試陣列的觀念，了解陣列的語法就能輕鬆解答。第三題與第四題測試C++的參數傳遞方式，只要了解傳值呼叫(Call by Value)、傳位址呼叫(Call by Address)與傳參考呼叫(Call by Reference)的差異度就能解答。
考點命中	第一題：《高點·高上程式設計概要講義》第一回，許振明編撰，頁168-172。 第二題：《高點·高上程式設計概要講義》第一回，許振明編撰，頁93-97。 第三題：《高點·高上程式設計概要講義》第一回，許振明編撰，頁82-87。 第四題：《高點·高上程式設計概要講義》第一回，許振明編撰，頁82-87。

一、試問以下C++程式若輸入believe，則輸出結果各為多少？（25分）

```
#include<iostream>
using namespace std;
char* POP_A(char*d){
    char*p=d;
    for(;*p;p++)*p=*(p+1);
    return d;
}
char* POP_B(char*d){
    char*p=d;
    while(*p)p++;
    *(p-1)=0;
    return d;
}
int main(){
    char d[1024];
    cin>>d;
    POP_A(d);
    cout<<d<<endl;
    POP_A(d);
    cout<<d<<endl;
    POP_B(d);
    cout<<d<<endl;
    POP_B(d);
    cout<<d<<endl;
    return 0;
}
```

答：
believe
lieve
liev
lie

```
#include <iostream>
using namespace std;
```

【版權所有，重製必究！】

```
char *POP_A(char *d)    // 刪除字串前面的第一個字元
{
    char *p=d;
    for(;*p; p++){
        *p=*(p+1);
    }
    return d;
}
```

```
char *POP_B(char *d){ // 刪除字串後面的第一個字元
    char *p=d;
    while(*p) p++;
    *(p-1)=0;
    return d;
}
```

```
int main(){
    char d[1024];
    cin >> d;
    POP_A(d);
    cout << d << endl;
    POP_A(d);
    cout << d << endl;
    POP_B(d);
    cout << d << endl;
    POP_B(d);
    cout << d << endl;
}
```

二、試問下列C++程式碼的輸出為何？（25分）

```
#include <iostream>
using std::cout;
using std::endl;

int main()
{
    int a[10][20][30][40];
    cout<<"a="<<a[5][5]-a[0][0]<<endl;
    cout<<"b="<<&a[5][5]-&a[0][0]<<endl;
    cout<<"c="<<a[5]-a[0]<<endl;
}
```

答：

a=3150
b=105
c=100

```
#include <iostream>
using std::cout;
using std::endl;
```

```
int main(){
    int a[10][20][30][40];
    cout << "a=" << a[5][5]-a[0][0] << endl; // (5-0)*(20*30)+(5-0)*30=3150
```

```

cout << "b=" << &a[5][5]-&a[0][0] << endl;           // (5-0)*20+(5-0)*1=105
cout << "c=" << a[5]-a[0] << endl;                 // (5-0)*20=100
}

```

三、下列C++程式執行後，試問A、B、C值分別為多少？（25分）

```

int F2(int &f, int f1) {
    int sum=(f1 + f++);
    return sum;
}
void F1(int &a, int b, int* c) {
    for (int i=a, j=b; j>=0; i++, j--) {
        if (c[j]%2==0)
            c[j]=F2(a, b);
        else
            c[j]=F2(b, a);
    }
}
void main() {
    int A=5, B=3, C[4]={1,4,7,6};
    F1(A, B, C);
    system("pause");
}

```

答：

A=7

B=3

C陣列 {11, 10, 9, 8}

```

#include <iostream>
using namespace std;

```

```

int F2(int &f, int f1){           // 第一參數 call by reference, 第二參數 call by value
    int sum = (f1+f++);
    return sum;
}

```

```

void F1(int &a, int b, int *c){    // 第一參數 call by reference, 第二參數 call by value, 第三參數 call by
    address
    for(int i=a, j=b; j>=0; i++, j--){
        if(c[j]%2==0)
            c[j]=F2(a,b);
        else
            c[j]=F2(b,a);
    }
}

```

```

int main(){
    int A=5, B=3, C[4]={1,4,7,6};
    F1(A, B, C);
}

```

【版權所有，重製必究！】

四、試問下列C++程式碼逐一執行後，Value與list輸出結果各為多少？（25分）

```
void swap_ref(int &a, int &b) {
    int temp;
    temp=a;
    a=b;
    b=temp;
}
void swap(int a, int b) {
    int temp;
    swap_ref(a, b);
    temp=a;
    a=b;
    b=temp;
}
void main() {
    int Value=5, list[3]={1,2,3};
    swap(Value+list[2]++, ++list[0]);
    swap_ref(list[0], ++list[2]);
}
```

答：

Value: 5

list陣列: {5, 2, 2}

```
#include <iostream>
using namespace std;
```

```
void swap_ref(int &a, int &b){ // call by reference
    int temp;
    temp=a;
    a=b;
    b=temp;
}
```

```
void swap(int a, int b){ // call by value
    int temp;
    swap_ref(a,b);
    temp=a;
    a=b;
    b=temp;
}
```

```
int main(){
    int Value=5, list[3]={1,2,3};
    swap(Value+list[2]++, ++list[0]);
    swap_ref(list[0], ++list[2]);
}
```

【版權所有，重製必究！】